Performance Evaluation of Silicon Physically Unclonable Function by Studing Physicals Values

Zhoua Cherif Jouini, Jean-Luc Danger, Lilian Bossuet

Performance Evaluation of Physically Unclonable Function by Delay Statistics

Zouha CHERIF JOUINI[1,2], Jean-Luc DANGER[2], Lilian BOSSUET[1]
[1] Université de Lyon
CNRS, UMR5516, Laboratoire Hubert Curien 42 000 Saint-Etienne, France
[2]Institut TELECOM, TELECOM ParisTech,
CNRS LTCI, 46 rue Barrault 75 634 Paris, France.
<lilian.bossuet@univ-st-etienne.fr>
<{cherif,danger}@telecom-paristech.fr>

***Abstract*—This paper presents a novel approach to evaluate silicon Physically Unclonable Functions (PUFs) implemented in FPGAs and based on delay elements. The metrics studied to characterize the PUFs are Randomness, Uniqueness and Steadiness. They take advantage of the measured physical values of elementary component making up the PUF. The delay distributions provide the interest to quantify the PUF at the physical level rather than carrying out a lot of experiments to get the PUF IDs at logical level. An Arbiter PUF composed of identical chains has been considered as a test chip to evaluate the method with the proposed metrics. Experiments have been carried out on CYCLONE II FPGA and the corresponding results shows the intra-device performance of the studied PUF.**

Keywords: Physically Unclonable Function (PUF); Silicon PUF, PUF metrics, FPGA.

I. Introduction

A Physically Unclonable Function (PUF) is a function which returns a value characteristic of an integrated circuit (IC). This device signature can be used to control the local behaviour of an algorithm. For instance cryptographic applications take advantage of PUF for authentication or key generation purposes. The Silicon PUF outputs a response (or ID) which depends on a control word, called the "challenge". A simple device authentication is based on a "challenge/response pair" which is the association between a set of challenges and the responses returned by a PUF. Due to the dispersion of the manufacturing process, the response for a given challenge be different between PUFs. Among the variety of PUF, the Silicon PUF is certainly the simpler to design as it does not require any specific technology. There are two main classes of Silicon PUFs: the PUFs based on delay comparisons, composed of identical elements, and the PUFs exploiting the initial state of memory blocks. The first silicon PUF introduced by Gassend & al is the Arbiter PUF [1] which compares the delay between two identical control paths. The Arbiter PUF can be derived to XOR PUF suggested in [7], and Lightweight Secure PUF [6], which is a composition of Arbiter PUFs. The Ring Oscillator (RO) PUF introduced by Suh & al [7] is a set of ring oscillators pairs which are compared in frequency. Guajardo & al. introduced the SRAM PUF [2] which is linked with the state of the SRAM at power up. The Butterfly PUF [4] works as the SRAM PUF but the memory point is based on two Flip-flops. This paper deals with PUFs based on delay chain comparison as arbiter PUFs or RO PUFs.

To perform an efficient characterization of PUFs, at least three metrics are necessary: randomness, uniqueness and steadiness. The randomness gives an estimate of the imbalance between the number of IDs at '0' and the IDs at '1' for all the challenges. The uniqueness indicates the entropy between two PUFs, either in the same device (intra-uniqueness) or between devices (inter-uniqueness). The steadiness expresses the level of PUF reliability which is decayed by the noise coming from the measurement environment.

The classical methods to characterize the PUFs are to perform statistical tests as the ones proposed by Hori and al. in [3]. These methods consider the set of logical PUF IDs, hence they need a lot of trials in order to run a Monte-Carlo estimation method. Our proposed method is based on the measurement of the physical values, i.e. the delays or frequencies. The advantage of this method is that only the number of tests is linear with M, M being the number of elements composing the PUF. Moreover it can compare with model dispersion as for the Pelgrom model at the design stage.

The article is organized as follows: Section II presents the background of the arbiter PUF architecture, used as an example and base of probability calculation. Section III describes the theory justifying the PUF metrics. The experiments and the results are presented in Section IV. Finally, conclusions and perspectives are discussed in Section V.

II. Background

A. *Arbiter PUF*

The example structure of the Arbiter PUF is made up of M identical delay elements structured as a mini crossbar 2x2, as illustrated in Figure 1. A step input simultaneously triggers the two paths which are controlled by a control word C, or challenge. At the end of the two parallel paths, a flip-flop D is used to convert the analog delay difference between the paths to a digital value which represents the response ID. Although the two paths are built identically, due to their intrinsic CMOS variation, the delays of the two selected paths are different. Therefore, the Arbiter PUF is expected to output unique IDs to the device.

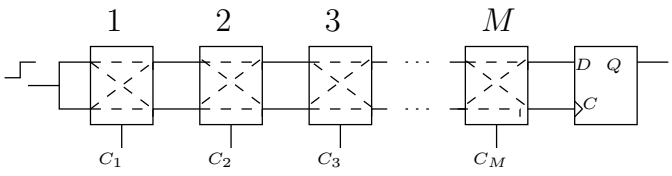

Figure 1. A structure of the Arbiter PUF

The arbiter PUF can be easily designed by using two delay chains of M elements as presented by Majzoobi in [5].

B. *Metrics based on Gaussian pdfs*

The base of the PUF metrics is to calculate a probability that expresses the quality to be random, unique or reliable. This probability is calculated from the delay distributions (or probability density function pdf) obtained by measurement. If we consider all the pdf as Gaussian, all the metrics need to know the probability to measure a value below a certain threshold. For instance the variable $x \in \mathcal{N}(\mu, \sigma^2)$, where μ is the mean and σ^2 is the variance. Then the probability to obtain a value of x less than a threshold t is given by :

$$Pr(x < t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{t} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \, \mathrm{d}x = \frac{1}{2}(1 + erf(\frac{(t-\mu)}{\sigma\sqrt{2}}) \quad (1)$$

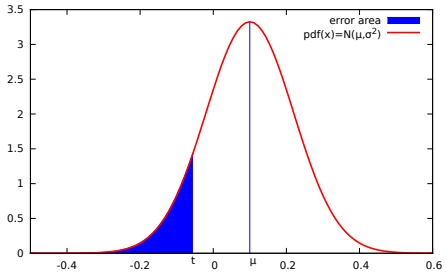

Figure 2. Error function and pdf(x).

III. Metrics Computation

In this section we define and explain the PUF metrics which are based on probability density functions of the measured delay. We consider an arbiter PUF which is made up of two delay chains of M delay elements d^i as in [5]. The notation used in this paper is listed in Table I.

Table I
NOTATION USED IN THE PAPER

Notation	Explanation
M	Number of elements in the PUF.
L	Number of studied PUFs.
T	Number of carried tests to evaluate the Steadiness of a PUF.
c_i	Challenge bit of the i^{th} element of the PUF. If $c_i = 0$, the path is the top else the bottom
d_0^i	Delay difference between top and bottom path, $c_i = 0$.
d_1^i	Delay difference between top and bottom path, $c_i = 1$.

A. *Randomness*

As we have seen before, the randomness represents the ability of the PUF to produce 0 or 1 with the same probability. An expression of the randomness in the probability domain can be:

$$Randomness = 1 - |Pr(ID = 0) - Pr(ID = 1)| \quad (2)$$

Therefore a randomness of 100% means the PUF ID states have the same probability of $1/2$. Considering the 2^M challenges, the probabilities to obtain an ID at 0 and 1 are:

$$Pr(ID = 0) = 1 - Pr(ID = 1) = Pr(\sum_{i=1}^{M} d_{c_i}^i \ < \ 0)$$

By considering two complementary challenges (i.e. one with c_i and the other with $\overline{c_i}$) we notice that:

$$\sum_{i=1}^{M} d_{c_i}^i \ + \ \sum_{i=1}^{M} d_{\overline{c_i}}^i \ = \ \sum_{i=1}^{M} (d_0^i + d_1^i)$$

Hence the mean value of the distribution D_R which represents the pdf of $\sum_{i=1}^{M} d_{c_i}^i$ is:

$$E(D_R) \ = \ 1/2 \sum_{i=1}^{M} (d_0^i + d_1^i). \quad (3)$$

As explained in Subsection II-B, the randomness computation can be done by using the distribution D_R illustrated in Figure 3 with $D_R = \mathcal{N}(E(D_R), M \cdot \sigma^2)$. This pdf is build with the measurement of the constant $\sum_{i=1}^{M} (d_0^i + d_1^i)$, and the variance $M \cdot \sigma^2$.

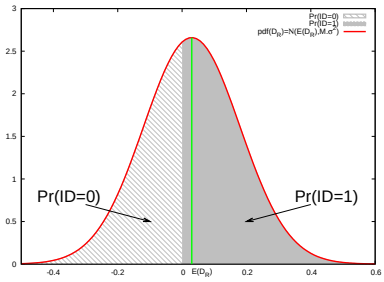

Figure 3. Randomness with the D distribution.

From Equation (1), we can derived:

$$Pr(ID = 0) = Pr(D_R < 0) = \frac{1}{2}(1 \ - \ erf(\frac{E(D_R)}{\sigma\sqrt{2 \cdot M}}))$$

Thus, from Equation (2) the randomness expression is:

$$Randomness = 1 - |erf(\frac{E(D_R)}{\sigma\sqrt{2 \cdot M}})|. \quad (4)$$

B. Uniqueness

The uniqueness is the ability of the PUF to behave differently than PUFs in another device (Inter-Uniqueness) or in the same device (Intra-Uniqueness). If we consider L PUFs, the global normal distribution D has $M \cdot L$ elements. We propose to compare the M distributions $D_i^L,\ i \in [1, M]$) of the delay difference $(d_0^i - d_1^i)$ of the L elements in the same range i, with the global distribution D. Hence if some elements i are biased, they will mitigate the comparison between their respective distribution D_i^L and the global distribution D. The uniqueness value is the mean of the M probabilities corresponding to M comparisons of distributions:

$$Uniqueness \ = \ \frac{1}{M} \sum_{i=1}^{M} Pr(D_i^L = D). \tag{5}$$

The compared distributions are considered normal, $D_i^L \in \mathcal{N}(\mu_i, \sigma_i^2)$ and $D \in \mathcal{N}(\mu, \sigma^2)$. The comparison of two Gaussian distributions can be expressed by the common area of these two distributions as illustrated in Figure 4 .

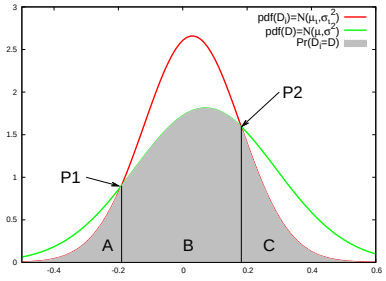

Figure 4. Example of two distributions comparison

The common area gives $Pr(D_i^L = D)$ which can be calculated by knowing the intersection points $P_{1,i}$ and $P_{2,i}$ and consequently the 3 areas A, B and C which are computed using Equation (1). The x-coordinate of the two intersection points $P_{1,i}$ and $P_{2,i}$ (of the two normal distributions are:

$$P_{1,i} = \frac{-(\mu\sigma_i^2 + \mu_i\sigma^2) - \sqrt{(\mu\sigma_i^2 + \mu_i\sigma^2)^2 - (\sigma_i^2 - \sigma^2)[\mu^2\sigma_i^2 - \mu_i^2\sigma^2 - 2\sigma^2\sigma_i^2 Ln(\sigma_i/\sigma)]}}{\sigma_i^2 - \sigma^2}$$

$$P_{2,i} = \frac{-(\mu\sigma_i^2 + \mu_i\sigma^2) + \sqrt{(\mu\sigma_i^2 + \mu_i\sigma^2)^2 - (\sigma_i^2 - \sigma^2)[\mu^2\sigma_i^2 - \mu_i^2\sigma^2 - 2\sigma^2\sigma_i^2 Ln(\sigma_i/\sigma)]}}{\sigma_i^2 - \sigma^2}$$

Equation (6) gives $Pr(D_i^L = D)$ with $\sigma_i > \sigma$ and $P_{1,i} < P_{2,i}$.

$$Pr(D_i^L = D) = 1 + \frac{1}{2}(\text{erf}(\frac{P_{1,i} - \mu}{\sqrt{2}\sigma}) - \text{erf}(\frac{P_{2,i} - \mu}{\sqrt{2}\sigma})) \tag{6}$$
$$+ \ \frac{1}{2}(\text{erf}(\frac{P_{2,i} - \mu_i}{\sqrt{2}\sigma_i}) - \text{erf}(\frac{P_{1,i} - \mu_i}{\sqrt{2}\sigma_i})),$$

C. Steadiness

The steadiness property of a PUF should show its ability to produce basically T times the same output, when using the same challenge on the same environmental conditions (temperature, voltage and noise).

Every delay difference of element i, $(d_0^i - d_1^i)$, is measured T times. The M distribution D_i^T of the T measured values are considered to be normal, centered in $E(d_0^i - d_1^i)$ with a variance S^2 identical for every element. The global distribution D corresponds to the distribution of mean values $E(d_0^i - d_1^i)$, centered in 0 (ideal randomness) with a variance σ^2.

Indeed, as much as we have delay difference near to 0, greater is the probability that the PUF ID is erroneous. If σ^2 is much greater than S^2, the computation of a steadiness metrics based on probability is greatly facilitated. If the delays are in the area $[-\lambda, \lambda]$ as shown by Figure 5, the distribution D can be considered uniform. λ is chosen in such a way that outside the $[-\lambda, \lambda]$ window the error probability is null. For instance, with $\lambda = 3S$, the distributions D_i^T have a confidence interval of 99.7% when in worse case (i.e. D_i^T centered in 0).

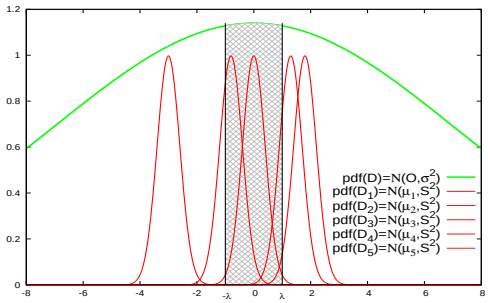

Figure 5. distributions after T measurements

Hence the total error probability is the product of the error probability when the delay is in the λ window multiplied by the probability of the delay in the λ window. This is expressed by the following equation:

$$Pr(error) = Pr(error \mid delay < |\lambda|) \cdot Pr(delay) < |\lambda|).$$

The steadiness is merely the opposite of this probability:

$$Steadiness = 1 - Pr(error). \tag{7}$$

The probability $Pr(error \mid delay < |\lambda|)$ is an integral of the *erf* function which can be approximated with the Taylor's series at the third order:

$$Pr(error \mid delay < |\lambda|) = \frac{1}{2\lambda} \int_0^{\lambda} (1 - \text{erf}(\frac{x}{S\sqrt{2}}))dx \tag{8}$$
$$= \frac{1}{6S} \int_0^{3S} (1 - \frac{2}{\sqrt{\pi}}(\frac{x}{S\sqrt{2}} - \frac{x^3}{3(S\sqrt{2})^3}))dx = \frac{4\sqrt{2\pi} - 3}{8\sqrt{2\pi}}$$

Using Equation (1), the probability that a delay is between $-\lambda$ and λ is:

$$Pr(delay \ < |\lambda|) \ = \ \text{erf}(\frac{\lambda}{\sigma\sqrt{2}}). \tag{9}$$

Therefore, using the Taylor's series of erf at the fist order (as $S << \sigma$), the steadiness has the following expression which depends merely on the ratio S/σ:

$$Steadiness \ = \ 1 \ - \ \frac{12\sqrt{2\pi} - 9}{8\pi} \times \frac{S}{\sigma}. \tag{10}$$

IV. EXPERIMENTS

A. Implementation

Tests have been carried out in a CYCLONE II EP2C35F672 with L=16 PUFs. The evaluated PUF is an arbiter PUF based on two parallel delay chains of M = 8 elements. The placement/routing of the 32 delay chains has been constrained to obtain the exact replication of the same chain. This is possible in ALTERA devices by using the same column and different rows for the IPs to replicate. The timing identities between the chains have been confirmed by the Time Quest timing Analyzer. The delay chains are closed to form a ring oscillator when in characterization phase. Figure 6 shows the architecture to evaluate each of the 16 arbiter PUFs.

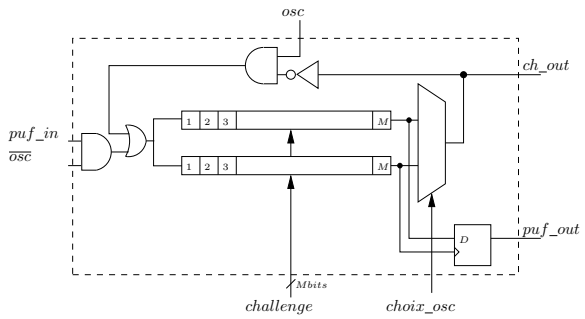

Figure 6. Implementation design

B. Measurement method

The ring oscillator output ch_out drives a counter of r bits. Concurrently the system clock period T_{clk} drives another counter which is sampled at the value n by the ch_out counter. The relation given by: $2^{r-1}T_{ch\_out} = nT_{clk}$, is used to derive T_{ch_out} from the value n. The measurement precision is of ${}^{1}/_{2^{r-1}T_{clk}} = 0.61ps$ with $r = 16$, $T_{clk} = 20ns$.

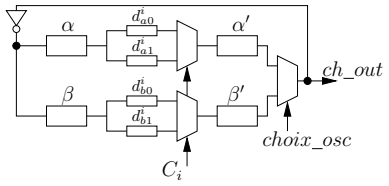

Figure 7. Measurement of element i

To measure the delay differences $(d_0^i - d_1^i)$ of every element i, the challenge bit C_i and the control signal $choix\_osc$ are driven alternately, the others challenge bits remaining constant. Figure 7 shows the delays which are involved in the measurements associated to the 4 combinations of C_i and $choix\_osc$. As the measurement is differential ($d_0^i - d_1^i$ = d_{a0}^i - d_{a1}^i - d_{b0}^i + d_{b1}^i), the external delays, as α and β, are eliminated. Concerning the randomness, the measurement of $E(D_R)$ expressed in 3 is global, the challenge bits are alternately all at '1' and all at '0'. We consider that the variance σ^2 of $d_0^i - d_1^i$ is equal to the variance of $d_0^i + d_1^i$ needed for the randomness evaluation. $T = 128$ experiments are carried out to measure the noise variance S^2 needed for the steadiness.

C. Results

The evaluation of the arbiter PUF based on two delay lines is presented in Table II. The PUF has the optimum quality when the metrics, equivalent to probabilities, are at 100%.

Table II
THE EXPERIMENTAL RESULTS OF THE INTRA-DEVICE EVALUATION OF THE ARBITER PUF

Performance indicator	Result
Randomness	0%.
Intra-Uniqueness	97.73%.
Steadiness	99.07%.

Table II reveals that the implemented PUF is absolutely not random (0%). This shows that the bias of the two independent delay lines has a lot of impact, as explained in [5], where a delay is introduced to compensate this bias. The arbiter PUF has good intra-Uniqueness and steadiness properties. However the steadiness should be estimated in other temperature and voltage configurations to cover all the conditions. The steadiness metrics gives a good idea of the capacity of the necessary error correction code to enhance the reliability towards 100%.

V. CONCLUSION

This paper presents a method to evaluate the silicon PUF based on delay chains or oscillators. The three proposed metrics which are randomness, uniqueness and steadiness are probabilities which have been formally expressed. Tests have been carried out on 16 arbiter PUFs based on two delay-chains. The results underline the weakness of this PUF implementation concerning the randomness but its strength for uniqueness end steadiness. The results should be further compare with other methods based on the logical values of the PUF. As other perspectives, the inter-uniqueness ((with different devices) will be studied and the steadiness will consider larger temperature and voltage conditions.

REFERENCES

[1] Blaise Gassend, Dwaine E. Clarke, Marten van Dijk, and Srinivas Devadas. Silicon physical random functions. In *ACM Conference on Computer and Communications Security*, pages 148–160, 2002.

[2] Jorge Guajardo, Sandeep S. Kumar, Geert Jan Schrijen, and Pim Tuyls. FPGA Intrinsic PUFs and Their Use for IP Protection. In *CHES*, Lecture Notes in Computer Science, pages 63–80. Springer, 2007.

[3] Yohei Hori, Takahiro Yoshida, Toshihiro Katashita, and Akashi Satoh. Quantitative and statistical performance evaluation of arbiter physical unclonable functions on fpgas. *Reconfigurable Computing and FPGAs, International Conference on*, 0:298–303, 2010.

[4] Sandeep S. Kumar, Jorge Guajardo, Roel Maes, Geert Jan Schrijen, and Pim Tuyls. The Butterfly PUF: Protecting IP on every FPGA. In Mohammad Tehranipoor and Jim Plusquellic, editors, *HOST*, pages 67–70. IEEE Computer Society, 2008.

[5] M. Majzoobi, F. Koushanfar, and S. Devadas. Fpga puf using programmable delay lines. In *Information Forensics and Security (WIFS), 2010 IEEE International Workshop on*, pages 1 –6, 2010.

[6] Mehrdad Majzoobi, Farinaz Koushanfar, and Miodrag Potkonjak. Lightweight secure pufs. In *Proceedings of the 2008 IEEE/ACM International Conference on Computer-Aided Design*, ICCAD '08, pages 670–673, Piscataway, NJ, USA, 2008. IEEE Press.

[7] G. Edward Suh and Srinivas Devadas. Physical unclonable functions for device authentication and secret key generation. In *DAC*, pages 9–14, 2007.